Cover Letter

This is a major revision of the extended version of our conference paper "Modeling The Temporally Constrained Preemptions of Transient Cloud VMs" that appeared at ACM HPDC 2020.

Compared to the preliminary conference version, we had made significant new contributions to algorithms, system design and implementation, and experimental evaluation. We first present a summary of the changes that we have made in response to the very helpful reviewer comments. We then describe the changes with respect to the conference version

1 Summary of changes in response to reviewer comments

Our sincere thanks to all reviewers for their careful, insightful, and very helpful feedback and comments. Below, we summarize the major changes. In the attached response, we describe how we have addressed every comment (in bold), along with locations of the relevant changes in the paper text.

Summary of changes by section:

1. New introduction to highlight the contributions
2. Merged background and related work
3. New data analysis (2 graphs) added, subsection on model generalization,
4. Made workload and bags of jobs assumption clear. We do not need exact job running times for our policies.
5. Implementation role of bags of jobs, and hot-spares clarification.
6. Added details of experimental conditions, exosphere, and HPC waiting times.
7. Added discussion section
8. Added conclusion

2 New Technical Contributions

The preliminary conference version introduced the notion of temporally constrained preemptions and a bathtub failure model. In this paper, we extend the model and its uses in significant ways, and address the implementation challenges by designing and implementing a new system, **SciSpot**. Our contributions are centered around a deeper theoretical analysis of our model, and the design and implementation of a system for running scientific computing applications on low-cost preemptible VMs. Specifically, we make the following *new* technical contributions (listed in the order of appearance in the paper):

1. A new "bag of jobs" abstraction for modern scientific computing workloads is introduced, which provides a new workflow model *and* opportunities for cost and performance optimization policies. We provide motivation and examples in Section 2.3; and show how these can be used for implementing policies at the end of Section 5.

2. We analyze our preemption model using reliability theory principles in Section 3.2.2. Specifically, we provide a formulation of the expected lifetime and hazard rate for our temporally constrained preemption model.

3. **A significant new theoretical contribution is our analysis of constrained preemptions using statistical mechanics principles.** In Section 3.3, we state and prove an important lemma which provides theoretical underpinnings of the bathtub shape of temporally constrained preemptions. Our phenomenological model connects statistical mechanics and transient computing in a new way, and provides the foundation for a deeper understanding of cloud resource management policies.

4. We present a new Transition-Points based job-scheduling and VM-reuse policy in Section 4.2.

5. We develop cost-minimization techniques using our preemption model in Section 4.3, which considers both the job resource requirements, and the cost and failure rates of different preemptible VMs in the cloud.

6. We implement and integrate all policies into a new system for scientific computing called SciSpot, described in Section 5.

3 New Empirical Evaluation

This paper also adds empirical evaluation of the new techniques as well as extended evaluation of the entire system. In particular:

1. We analyze the job-scheduling policies in Figure 5.

2. Impact of VM selection, and analysis of running times of different scientific computing applications, in Section 6.2 and Figure 10.

3. Cost analysis and evaluation in Section 6.3.
4. Comparison with HPC waiting times in Section 6.4.
5. Comparison of SciSpot cost with the state of the art in transient server selection (ExoSphere), in Section 6.3.